REMARKS

Claims 1-21 are pending in the application. In response to the Office Action, applicants have amended independent claims 1, 3, 11, and 12. Claims 1-21 remain in this application and are now pending for reconsideration.

Claim 1 was rejected under 35 U.S.C. § 102(b) as being anticipated by Lindholm et al. (U.S. Patent No. 5,797,004). Claims 2-3 and 9-10 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lindholm et al. in view of Spix et al. (U.S. Patent No. 5,179,702). Claims 4-7 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lindholm et al. in view of Kerrigan et al. (U.S. Patent No. 5,404,488). Claims 8, 11-12 and 17-19 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lindholm et al. in view of Spix et al. in view of Tillier (U.S. Patent No. 6,421,742). Claims 13-16 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lindholm et al. in view of Spix et al. in view of Tillier in view of Kerrigan et al. Claims 20-21 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lindholm et al. in view of Spix et al. in view of Kerrigan et al.

Lindholm et al. discloses an object-oriented approach for control of a shared resource. Access to an object is provided through a set of methods embedded within the object. Some methods may be synchronized to limit access to the resource by multiple threads. Access to a cache manager is managed by a dedicated synchronization construct so that only one thread of execution can access the cache manager at one time, and all subsequent synchronization construct manager requests are put on a waiting thread list until the first thread completes use of the cache manager. Synchronization constructs are used to synchronize objects with threads using well known synchronization constructs such as mutexes, monitors, and semaphores. A mutex method limits access to a shared resource to a single thread, a monitor method suspends one or more threads until the data accessed by the method becomes available, and a semaphore method allows up to N threads simultaneous access to a shared resource. Lindholm et al. discloses a method for caching and allocating the thread synchronization constructs, but does not disclose specifics about the type of synchronization that may be employed by a method accessing the shared resource. Lindholm et al. does not disclose synchronizing multiple worker threads of operation with a single update thread to achieve mutual exclusion between the multiple worker threads performing work on the shared resource and a single update thread updating or changing the state of the shared resource as claimed in independent claim 1, for example. The mutex disclosed in Lindholm et al. is locked when it is allocated and has its synchronizers list (or synchronizer identifier) updated to identify the thread that is synchronized with an object and unlocked when it is de-allocated. This is different than the present invention as claimed, which allows processors to perform work on a shared resource concurrently while supporting state changes or updates of the shared resources using a synchronization algorithm to synchronize multiple worker threads of operation with a single update thread. As discussed above, a mutex method limits access to a shared resource to a single thread.

The Examiner has asserted that the synchronization algorithm of Lindholm et al. locks a shared resource. The Examiner has asserted that a cache management software module handles allocation and deallocation of shared resources. However, the Examiner has not specifically pointed out how the cache management software is analogous to the single update thread in the presently claimed application, or how that cache management software module achieves mutual exclusion between multiple worker threads and a single update thread changing the state of the shared resource without requiring serialization of all threads. Rather, the portions of the Lindholm et al. patent pointed out by the Examiner refer to synchronizing objects with one or more threads (for example, column 5, lines 34-34 and column 6, lines 10-21).

The Lindholm et al. patent does not disclose a synchronization method as set forth in the claims of the present application. In addition, it would not have been obvious to combine the references as asserted by the Examiner, except in hindsight in view of the present application. Even assuming that one of ordinary skill in the art would have been motivated to combine the references relied upon by the Examiner, at best, one might come up with a system with multiple worker threads and a shared resource in which only one worker thread worked on the resource at a time, or a system with one master thread executing an application to generate data for respective slave threads. This is different than the invention claimed in the present application.

The Applicants respectfully traverse the rejections in the outstanding Office Action for at least the reasons set forth above.

In view of the foregoing, the application is considered to be in condition for allowance. Early notification of the same is earnestly solicited. If there are any questions regarding the present application, the Examiner is invited to contact the undersigned attorney at the telephone 503-439-8778.

Respectfully submitted,

February 14, 2005
Date

Mark C. Van Ness
Reg. No. 39,865

Blakely Sokoloff Taylor & Zafman
12400 Wilshire Boulevard, Seventh Floor
Los Angeles, CA 90025

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313 on:

2/14/05
Date of Deposit

Ann P Pearson
Name of Person Mailing Correspondence

2/14/05
Signature Date